Dense visual mapping of large scale environments for real-time localisation

Maxime Meilland, Andrew I. Comport, Patrick Rives

Dense visual mapping of large scale environments for real-time localisation

Maxime Meilland, Andrew Ian Comport and Patrick Rives

***Abstract*— This paper presents a method and apparatus for building dense visual maps of large scale 3D environments for real-time localisation and navigation. A spherical ego-centric representation of the environment is proposed that is able to reproduce photo-realistic omnidirectional views of captured environments. This representation is novel in that it is composed of a graph of locally accurate *augmented spherical panoramas* that allows to generate varying viewpoints through novel view synthesis. The spheres are related by a graph of 6dof poses which are estimated through multi-view spherical registration. To acquire these models, a multi-baseline acquisition system has been designed and built which is based on an outward facing ring of cameras with diverging views. This configuration allows to capture high resolution spherical images of the environment and compute a dense depth map through a wide baseline dense correspondence algorithm. A calibration procedure is developed for an outward facing camera ring that imposes a loop closing constraint, in order to obtain a consistent set of extrinsic parameters. This spherical sensor is shown to acquire compact, accurate and efficient representations of large environments and is used for real-time model-based localisation.**

I. Introduction

Acquiring 3D models of large scale environments is currently a key issue for a wide range of applications ranging from interactive personal guidance devices to autonomous navigation of mobile robots. In these applications it is important, not only for human operators but also for autonomous robots, to maintain a world map that holds a rich set of data including photometric, geometric and saliency information. It will be shown in this paper why it is advantageous to define an *ego-centric* representation of this information that allows fast model acquisition whilst maintaining optimal realism and accuracy.

Clearly, an a-priori 3D model simplifies the localisation and navigation task since it allows to decouple the structure and motion estimation problems. Current state of the art approaches mostly rely on global 3D CAD models [10] that are based on tools and representations that been developed mainly for texture mapped virtual reality environments. Unfortunately, these representations have difficulty in maintaining true photo-realism and therefore introduce reconstruction errors and photometric inconsistencies. Furthermore, these models are complicated to acquire and often resort to heavy off-line modelling procedures. Whilst efforts are being made to use sensor acquisition systems that automatically acquire these classical virtual 3D models [10], it is suggested in this paper that they are not sufficient to precisely represent real-world data. Alternatively, it is proposed to use an ego-centric model [18] that represents, as close as possible, real sensor measurements.

A well known ego-centric representation model for camera sensors is the spherical panorama. Multiple cameras systems such as in [2] allow construction of high resolution spherical views via image stitching algorithms such as reviewed in [21]. However, contrary to virtual reality models, these tools have been developed mainly for qualitative photo-consistency but they rarely require 3D geometric consistency of the scene. This is mainly due to the fact that, in most cases, it is impossible to obtain 3D structure via triangulation of points when there is no or little baseline between images. Another approach is to use a central catadioptric omnidirectional camera [20] and warp the image plane onto a unit sphere using the model given in [9]. Unfortunately, that kind of sensor has a poor and varying spatial resolution and therefore is not well adapted to a visual memory of the environment. Furthermore, these approaches assume a unique center of projection, however, manufacturing such a system is still a challenging problem [16].

In order to take advantage of both 3D model based approaches and photometric panoramas it is possible to *augment* the spherical image with a depth image containing a range for each pixel. An augmented sphere then allows to perform novel view synthesis [1], [7], [18] in a local domain in all directions. There are many approaches for obtaining depth information ranging from laser range finders to structured light and stereo matching with triangulation. Laser approaches [8], [6] are expensive and cumbersome and structured light RGB-D systems [11] are short range and only work indoors. In [17] a spherical camera is built by combining two fish-eye lenses with a mirror to project both images onto a unique sensor. Likewise, in [5], stereo vision tracking is performed using four omnidirectional mirrors. This type of sensor has a delicate calibration process and again has uneven spatial resolution. A recent work, [14] uses two rotating line scan cameras to acquire image spheres at different heights. Stereo is then achieved via dense matching between the spheres, however, this system is not adapted to a vehicle in motion due to the slow acquisition frequency of the rotating cameras. Multi-camera systems, however, can perform dense stereo-matching [12], which can be performed outdoor and indoor, which provides high spatial resolution, corresponding depth and photometric data.

These ego-centric models are, however, local and do

M. Meilland and P. Rives are with INRIA Sophia Antipolis Méditerranée, 2004 Route des Lucioles BP 93, Sophia Antipolis, France, {name.surname}@inria.fr

A.I. Comport is with CNRS, I3S Laboratory, Université Nice Sophia Antipolis, 2000 Route des Lucioles BP 121, Sophia Antipolis, France, comport@i3s.unice.fr

not provide a global representation of the environment. This problem can be solved by considering multiple augmented spheres connected by a *graph* of poses that are positioned optimally in the environment. Simple spherical images positioned in the environment are already found in commercial applications such as Google Street View, and more recently [15]. The easiest method for positioning spheres would be via a global positioning system (GPS), however, in urban environments this system fails easily due to satellite occlusion. Alternatively, the robot-centered representation introduced in [18] positions augmented views globally within a precise topological graph via accurate stereo visual odometry [7] and does not require any external sensor. The present paper extends this preliminary work.

A. Contribution

In this paper, a custom made multi-camera spherical imaging system is presented that deviates from classic spherical sensor in that there is a baseline between each camera. The new system is designed to maximise the overlap among six wide field of view cameras equally placed on an hexagon. A technique is provided for calibrating this outward looking ring of stereo cameras with a loop closing constraint. This system is then shown to simultaneously extract a dense depth-map between all stereo pairs using wide-baseline dense matching [12]. This dense depth-map is then blended and mapped onto a unit sphere with 3D geometric constraints. Spheres are placed optimally within a global graph based on a robust statistic criteria. The full collection of spheres is stored in a GIS (Georeferenced Information System), which is then used during the navigation phase. This ego-centric visual memory is then shown to be used for real-time robust localisation with respect to different online visual sensors (webcam, monocular, stereo). The main advantages of this spherical representation are :

- An ego-centric representation allows to maintain accurate local sensor data (i.e. photometric consistency) and only provides the necessary information (e.g. locally around navigation path).
- Augmenting photometric spherical panorama's with dense depth allows to perform local novel view synthesis.
- A spherical representation provides all local view directions and therefore allows combination of different kinds of sensors like perspective cameras, multi-view cameras or omnidirectional cameras and laser range finders.
- Full-view sensors well condition the observability of 3D motion [2] which greatly improves robustness.
- Can be made invariant to illumination variation as in [19]

II. Real-time Ego-centric Tracking

As mentioned in the introduction, the objective of this work is to perform real-time tracking using a known environment model (see Fig. 1). The essential part of this paper is therefore divided into two distinct but inter-related aspects:

- **Learning** - This phase consists in acquiring a 3D model of the environment and representing this information in an optimal manner for "on-line" localisation. It has been chosen to develop a learning approach that is also *efficient* so that, firstly, in a practical sense environments can be acquired rapidly and secondly, so that the approach may be used for online mapping in the near future. Essentially this involves filming, tracking and mapping the 3D environment ($\approx$1Hz depending on the approach). See Section III for the local ego-centric 3D model and its acquisition system along with Section IV for the global graph learning.
- **Online tracking** - This real-time phase involves estimating the 6 d.o.f. pose of one or several camera(s) at frame-rate (here 45 Hz). This phase must take into account efficient optimisation techniques that require a maximum amount of computation to be performed "off-line" during the learning phase. See Section V-B.

III. Spherical ego-centered model

An ego-centric 3D model of the environment is defined by a graph $\mathcal{G} = \{\mathbf{S}_1, \ldots, \mathbf{S}_n; \mathbf{x}_1, \ldots, \mathbf{x}_m\}$ where $\mathbf{S}_i$ are *augmented spheres* that are connected by a minimal parametrisation $\mathbf{x}$ of each pose as:

$$\mathbf{T}(\mathbf{x}) = e^{[\mathbf{x}]_\wedge} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \in \mathbb{SE}(3), \tag{1}$$

where $\mathbf{x}^{ab} \in \mathbb{R}^6$ is the 6 d.o.f. twist between the sphere a and b (see Fig. 1) defined as:

$$\mathbf{x} = \int_0^1 (\boldsymbol{\omega}, \boldsymbol{v}) dt \in \mathfrak{se}(3), \tag{2}$$

which is the integral of a constant velocity twist which produces a pose $\mathbf{T}$. The operator $[.]_\wedge$ is defined as follows:

$$[\mathbf{x}]_\wedge = \begin{bmatrix} [\boldsymbol{\omega}]_\times & \boldsymbol{v} \\ \mathbf{0} & 0 \end{bmatrix}, \tag{3}$$

where $[.]_\times$ represents the skew symmetric matrix operator.

A. Augmented visual sphere

Each sphere is defined by the set $\mathbf{S} = \{\boldsymbol{\mathcal{I}}_s, \boldsymbol{\mathcal{P}}_s, \mathbf{Z}_s, \mathbf{W}_s\}$ where

- $\boldsymbol{\mathcal{I}}_s$ is the photometric spherical image. This image is obtained from the custom camera system presented in Section III-B by warping multiple images onto the sphere as will be defined in Subsection III-C.
- $\boldsymbol{\mathcal{P}}_s = \{\mathbf{q}_1, \ldots, \mathbf{q}_n\}$ is a set of evenly spaced points on the unit sphere where $\mathbf{q} \in S^2$. These points have been sampled uniformly on the sphere as in [18].
- $\mathbf{Z}_s$ are the depths associated with each pixel which have been obtained from dense stereo matching as will be detailed in Section III-E. The 3D point is subsequently defined in the sphere as $\mathbf{P} = (\mathbf{q}, \mathbf{Z})$.
- $\mathbf{W}_s$ is a saliency image which contains knowledge of good pixels to use for tracking applications. It is obtained by analysing the Jacobian of the warping function so that the pixels are ordered from best to worst in terms of how they condition the pose estimation problem (the interested reader can see [18] for detail).

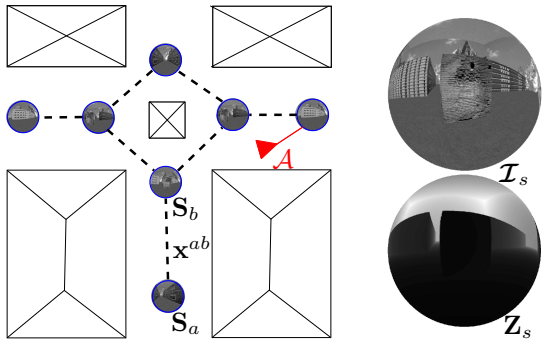


Fig. 1. Ego-centric representation: graph of spheres $\mathcal{G}$ and augmented spheres containing grey levels and their corresponding depths projected on a unit sphere $\mathcal{S}^2$. $\mathcal{A}$: an agent (robot or person) is shown connected to the graph.

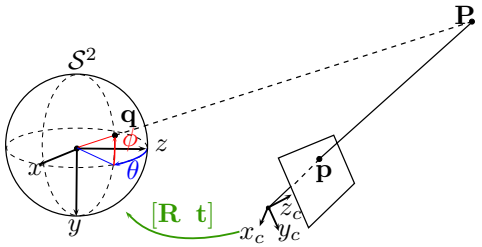


Fig. 2. The spherical projection of a 3D point $\mathbf{P}$ on a sphere. $\mathbf{R}, \mathbf{t}$ is the rigid transformation between the camera and the sphere.

B. Spherical acquisition system

Following on from the introduction, no commercial camera systems are yet available to acquire spherical panoramas with depth information that perform both outdoor and indoor whilst providing a high spatial resolution map of the environment. In that respect a new acquisition system has been designed that purposely maintains a significant baseline between *multiple divergent cameras*. The idea being to equally place the cameras in a ring configuration (see Fig. 5(a)). The advantage of this design is that the baseline between each pair of cameras allows to compute dense correspondences and their corresponding depth maps. One particularity that makes this system more original is the fact that the cameras are in a divergent configuration. Indeed most multi-baseline camera systems are configured so as to observe the same point(s) in 3D space. This new configuration therefore requires additional modelling to account for diverging views and loop closing constraints around the camera ring.

The particular implementation of the system constructed for this purpose is composed of six high resolution cameras (1292×964), each mounted with a wide angle lens (125^o) and configured in a hexagon. The use of wide field-of-view sensors ensures near-complete overlap between each pair of cameras and almost covers the full 360 degrees of the sphere.

C. Image warping: Novel view synthesis

To create a spherical panorama from a multi-baseline camera system it is necessary to warp and blend each camera's image onto the sphere (see Fig. 5(b)). For the purpose of this subsection, suppose that both intrinsic and extrinsic camera calibration has been achieved and that dense depth information has been determined (for each pixel). With this information, image warping (or novel view synthesis [1]) is achievable.

Whilst the warping function is presented here to warp each camera's image onto a spherical panorama, it will be defined in a general manner since it is also a key component for Section IV in defining the optimisation criteria for off-line pose estimation along with Section V-B for real-time tracking.

The geometric part of the warping function $w(.)$, is defined to represent the transfer of 3D points of an augmented sphere wrt. a current generic sensor (see Fig. 2) such that:

$$\mathcal{I}_S(\mathcal{P}_s) = \mathcal{I}\big(w(\mathbf{T}(\mathbf{x}), \boldsymbol{\xi}; \mathcal{P}_s, \mathbf{Z}_s)\big), \tag{4}$$

where $\mathcal{I}$ are the current sensor intensities measurement, $\boldsymbol{\xi}$ is the intrinsic parameter vector (based on the sensor type, e.g. perspective, catadioptric, spherical etc...) and $\mathbf{T}(\mathbf{x})$ is the rigid pose transformation (1) between sensors (extrinsic parameters). Since there is rarely a one-to-one pixel correspondence in $\mathcal{I}(\mathbf{p})$, corresponding intensities are interpolated at pixel location $\mathbf{p}$ (i.e. by bilinear interpolation). Since two cameras measure the same intensity (due to overlap) their values are fused using Laplacian blending [4]. This compensates exposure differences between cameras.

For a spherical camera the warping function is defined as:

$$\mathbf{q} = \frac{\mathbf{RP} + \mathbf{t}}{\|\mathbf{RP} + \mathbf{t}\|} \in \mathbb{S}^2. \tag{5}$$

D. Closed-loop Calibration of Diverging Cameras

As mentioned previously, a multi-baseline divergent camera system presents certain particularities in terms of intrinsic and extrinsic calibration as well as in terms of divergent views. In Fig. 3 it can be seen that even if the system contains multiple cameras, only pairs of cameras observe the same parts of the scene which means that the camera system is essentially composed of several stereo-pairs.

Since the calibration patterns are only viewed by two cameras simultaneously, standard multi-camera calibration techniques such as [22] are unfortunately not suitable. It is, however, possible to successively compute the extrinsic parameters of each pair of cameras, but in this case calibration parameters will not be completely consistent when combining poses around the loop. Therefore, it is proposed here to define the calibration problem with a global loop closing constraint that allows to further constrain the extrinsic parameters of the system so that the poses around the loop remain consistent.

The new extrinsic calibration procedure is modelled so as to simultaneously estimate pattern poses $\mathbf{x}_i^p$ and camera poses $\mathbf{x}_i^c$ with respect to a central coordinate system, where the pose vectors are defined in equation (2). The unknown state of the system is therefore defined as:

$$\mathbf{x}^\Sigma = (\mathbf{x}_1^c, ..., \mathbf{x}_M^c, \mathbf{x}_1^p, ..., \mathbf{x}_N^p)^\top \tag{6}$$

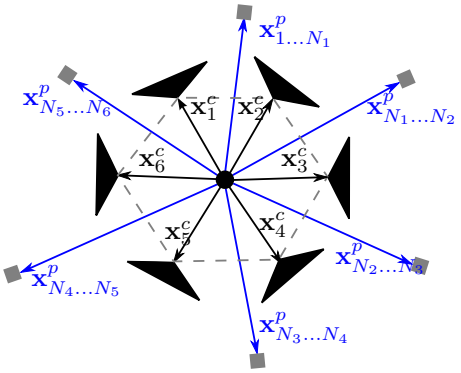


Fig. 3. Spherical system calibration. $\mathbf{x}^c$ are the camera pose parameters and $\mathbf{x}^p_N$ are the N patterns pose parameters.

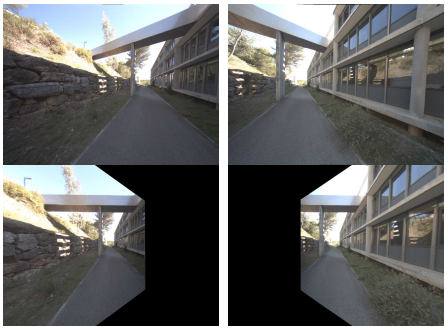


Fig. 4. Rectification of two divergent stereo images. *Top*: Un-rectified left and right images. *Bottom*: Rectified left and right images.

with N the number of calibration patterns observed over multiple snapshots and $M = 6$ is the number of cameras.

The global optimisation criteria is then defined (with abuse of notation) as the error between a vector of warped pattern's points $w(\boldsymbol{\mathcal{P}}_p)$ and a vector of matching points in the image $\boldsymbol{\mathcal{P}}_m$:

$$\mathbf{e}(i,j) = \boldsymbol{\mathcal{P}}_m - w\left(\mathbf{T}(\mathbf{x}^c_i)\mathbf{T}(\mathbf{x}^p_j), \boldsymbol{\xi}_i; \boldsymbol{\mathcal{P}}_p, \mathbf{Z}_p\right), \quad (7)$$

where i and j are the respective camera number and pattern number (see Fig 3) and $\mathbf{K}(\boldsymbol{\xi}_i) \in \mathbb{R}^{3\times3}$ the intrinsic parameters matrix of camera i. In this case, the warping function (4) is defined to represent the projection of the points of the pattern j on camera i.

For a perspective camera the warping function $w(.)$ is defined such that:

$$\mathbf{p} = \boldsymbol{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{P}, \quad (8)$$

where $\mathbf{P}$ is the 3D Euclidean point within the calibration pattern.

Using this error function, it is possible to find an optimal $\widehat{\mathbf{x}}^{\Sigma}$, by minimizing the re-projection error for each overlap and each pattern using:

$$\widehat{\mathbf{x}}^{\Sigma} = \arg\min_{\mathbf{x}^{\Sigma}} \sum_{i=1}^{6}\left(\sum_{j=1}^{N}\|\mathbf{e}(i,j)\|^2.\eta(i,j)\right) \quad (9)$$

$$\eta(i,j) = \begin{cases} 1 & \text{if pattern j is seen by camera i} \\ 0 & \text{otherwise} \end{cases}$$

Iteratively minimizing the cost function (9) allows to estimate each camera's pose $\mathbf{x}^c_i$ while respecting the loop closing constraint. In order to avoid local minima, the optimisation problem is initialised with stereo calibration and the intrinsic parameters $\boldsymbol{\xi}_k$ are not recomputed since locally they are already estimated accurately.

The calibration results of the system are shown on Fig. 5(b). It can be seen that if the extrinsic parameters are estimated independently, the error is accumulated (red positions) from the camera 1 to camera 6. By estimating the parameters in a global optimization, the loop is closed (blue positions). As a practical note, due to the scale of the system it is complicated to construct a single rigid 3D calibration pattern which surrounds all the cameras simultaneously such as [17]. Here, only a classical checker-board pattern, which was dimensioned to cover large parts of the image, has been used so as to constrain different depths.

E. Dense correspondence

In order to construct high resolution spherical images (approx. 4.5 million pixels) that are augmented with depth, it is necessary to perform dense matching. Although dense matching is not the aim of this paper, several difficulties were encountered due to the divergent wide-baseline acquisition system which has required a careful choice of algorithm and has highlighted potential problems. Firstly, classic dense matching across diverging views is a non-trivial problem due to the significant difference in resolution of the scene between two cameras. Secondly wide baseline stereo cameras allow to well constrain far off objects, however, they also require searching much larger intervals on the epipolar line.

In the system presented here, each camera's image half overlaps with neighbouring left and right cameras respectively. The major difficulty in this configuration is due to the hexagonal configuration, the angle between optical axes of two adjacent cameras are clearly divergent (60^o) and the baselines are (65cm) wide. This creates a significant difference in base image resolution of the scene and requires a large disparity search range (See Fig. 4).

In order to perform dense matching, each stereo pair is first rectified. The important advantage of rectification is that computing stereo correspondences is reduced to a 1-D search problem along the horizontal raster lines of the rectified images. The disadvantage being that the difference in resolution may produce approximation errors in the rectified image. Even with rectified images, the differences in resolution (due to perspective distortions) and illumination (due to shading correction) between two images produce erroneous dense matches with standard techniques. In this paper Semi-Global Block Matching [12] was used.

IV. GLOBAL SPHERE POSITIONING

Now that the elementary augmented spheres have been defined, this next section is dedicated to defining the complete

graph (defined in Section III) that makes up a 3D model. This will involve introducing a model for accurately estimating the edges (poses) that link the vertices together and also on how to optimally place the vertices (spheres) within the 3D environment.

A. Spherical visual odometry

To accurately recover the position of the spheres with respect to one-another, a 6 d.o.f. multi-camera localisation model is proposed based on accurate dense localisation [7], [18]. Considering $\mathcal{I}_S$, an augmented sphere defined in Section III-B, the objective is now to compute the pose between a reference sphere and the next one. The localisation problem (also known as visual odometry) is then to estimate the incremental pose $\mathbf{T}(\tilde{\mathbf{x}})$. Since this is a local optimisation approach it is assumed that the camera framerate is high (30Hz) and that interframe displacements are small ($\leq 2m$), meaning a maximum speed of $\sim 200km/h$.

It is noted here that dense visual odometry is computationally efficient and locally very accurate [7] so it has been deemed unnecessary to perform costly bundle adjustment on local visibility windows (although this slightly improves the estimate it makes timely scene acquisition practically unfeasible).

Using an iterative optimization scheme as given in the Appendix VIII-A, the estimate is updated at each step by an homogeneous transformation:

$$\widehat{\mathbf{T}} \leftarrow \widehat{\mathbf{T}}\mathbf{T}(\mathbf{x}), \tag{10}$$

where $\widehat{\mathbf{T}}$ is the current pose estimate with respect to the closest reference sphere which is determined from the previous iterations up to time $t-1$.

The error measure between a reference sphere and a spherical multi-view system is then defined as follows:

$$\mathbf{e}_i = \rho\left(\mathcal{I}_i\left(w\left(\mathbf{T}(\mathbf{x}_i^c)\widehat{\mathbf{T}}\mathbf{T}(\mathbf{x}); \boldsymbol{\mathcal{P}}_s, \mathbf{Z}_s\right)\right) - \mathcal{I}_s(\boldsymbol{\mathcal{P}}_s, \mathbf{Z}_s)\right), \tag{11}$$

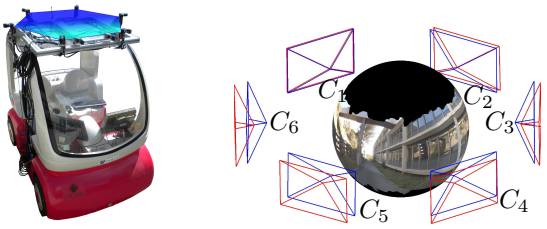
where $i = 1 \ldots 6$ is the camera index, $w(.)$ is the warping function of eq. (4), $\mathbf{x}_i^c$ are the corresponding extrinsic camera parameters obtained in III-D and the intrinsic parameters are assumed implicit, and ρ is a robust M-estimator given in [13] where the robust statistical weight is defined by the Huber weighting function.

B. Spherical node placement

Indeed, the vertices should be carefully placed in the world so as to represent the environment with little redundancy. One preliminary technique to achieve this goal locally is to observe criteria between an initially selected reference sphere and surrounding spheres. In practice, the trajectory of the acquisition system along a sequence is computed by integrating elementary displacements estimated from successive spherical registration. The strategy used here is to maintain as long as possible the reference sphere to minimize the drift introduced when a new reference sphere is taken. Therefore a new reference sphere is placed according to the Median Absolute Deviation (MAD) and the norm of the error:

$$\lambda_1 < Median(\mathbf{e} - Median(\mathbf{e})), \quad \lambda_2 < |\mathbf{e}| \tag{12}$$

where $\mathbf{e}$ is the error defined in (11). A new reference sphere is therefore placed when the MAD measure of the error is greater than a defined threshold, or when the weighted error norm is too large. Since the registration technique is direct, local precision on the topological graph is very good (around 1% drift), which is important for online navigation.



(a) (b)

Fig. 5. (a): Spherical system mounted on a Cycab robot. (b): Warping of images onto the sphere. For calibration, in red, the camera poses successively estimated between each overlap, we can see the drift when loop closing is not performed. In blue the cameras poses estimated with the loop closing constraint.

V. Results

A. Map Building

A 7364×6 image sequence was acquired over a 1500 meter long trajectory, using the custom spherical acquisition system mounted on a mobile robot 5(a). The environment contains corridors, near and far buildings, vegetation, parked cars, straight sections, corners and several hills (demonstrating the 6 d.o.f. trajectory), which well represent most aspects of an urban environment. Sphere construction and global positioning was computed off-line at around 1Hz.

Since the positioning method is based on visual odometry, small errors may be integrated leading to inconsistency in the global map. A Loop closure detection was performed and a global pose optimisation was used to correct the drift. Fig. 6(c) shows the final graph, composed of 310 augmented spheres, that cover the entire trajectory and well represent the robot path. Since the spheres are positioned using a dense direct method, the graph's edges are accurately estimated, making navigation between nodes continuous which allows interactive navigation within a 3D world by an end-user (see Fig. 6(b)).

B. Real-time tracking and localisation

It is considered that during online navigation, a current image $\mathcal{I}$, captured by a generic camera (e.g. monocular, stereo or omnidirectional) and an initial guess $\widehat{\mathbf{T}}$ of the current camera position are available. This initial guess permits the extraction of the closest reference sphere $\mathcal{S}$ from the graph. Since a sphere provides all local information necessary for 6 dof. localisation, an accurate estimation of the

pose is obtained by an efficient direct minimization, related to (11):

$$\mathbf{e} = \rho\left(\boldsymbol{\mathcal{I}}\left(w\left(\widehat{\mathbf{T}}\mathbf{T}(\mathbf{x}); \boldsymbol{\mathcal{P}}_s, \mathbf{Z}_s, \mathbf{W}_s\right)\right) - \boldsymbol{\mathcal{I}}_s(\boldsymbol{\mathcal{P}}_s, \mathbf{Z}_s, \mathbf{W}_s)\right), \tag{13}$$

where $\mathbf{W}_s$ is the saliency image [18] which selects only informative pixels for warping, which speeds up the algorithm without degrading observability and accuracy. The error function $\mathbf{e}$ is minimized using an iterative (IRLS) non-linear optimization detailed in Appendix VIII-A. A maximum amount of pre-computation is performed offline during the construction of the spheres (e.g. Jacobian matrices and saliency maps) allowing the online algorithm to be computationally efficient, which allows the camera pose to be estimated at frame rate.

To farther improve performance, a coarse-to-fine optimization strategy is employed by using multi-resolution spheres (e.g.. constructed by Gaussian filtering and subsampling [4]). The minimization begins at the lowest resolution and the result is used to initialize the next level repeatedly until the highest resolution is reached. This greatly improves the convergence domain/speed and some local minima can be avoided.

In order to choose the closest sphere for tracking within the graph, it necessary to define a metric. Contrary to non-spherical approaches, a sphere provides all viewing directions and therefore it is not necessary to consider the rotational distance (to ensure image overlap). The closest sphere is subsequently determined uniquely by translational distance. In particular this avoids choosing a reference sphere that has similar rotation but large translational difference which induces self occlusions of buildings and also differences in image resolution caused by distance (which affects direct registration methods).

The online algorithm was tested and validated on a subset of the full spherical graph containing 12 spheres. A real-time implementation has been realized in C++. Using only salient pixels, the online localisation runs at 45Hz on an Intel Core 2 Duo laptop. A vehicle equipped with a monocular camera of 800×600 pixels in size with a frequency of 45Hz, was moved within the neighbourhood of the graph.

The results of Fig. 6(a) show an overview of the estimated trajectory in green, (the black part (a) indicates a forward-reverse movement of the vehicle), with some camera poses. The camera starting point is $(X{=}{-}0.4, Y{=}{-}0.1, Z{=}0.8)$ and the vehicle begins to move in positive Z direction until the position $(X{=}{-}1.8, Y{=}0.6, Z{=}20.5)$. Then the robot is moved backward (black trajectory) until position $(X{=}3.3, Y{=}0.6, Z{=}18.8)$ to return to the initial position by reversing.

The proposed method was able to accurately track the camera at video frame rate, for a vehicle navigating in different directions, within a local region of the graph, which emphasizes the advantages of a spherical ego-centered representation [1]. Recent results have shown that this technique can be made robust to illumination changes [19].

[1]A high quality video is available at:
http://www-sop.inria.fr/arobas/videos/Globeye/DenseVisualMappingHQ.mp4

VI. CONCLUSIONS

The approach described in this paper allows reconstructing dense visual maps of large scale 3D environments. It has been shown that this representation is capable of reproducing photometrically accurate views locally around a learnt graph. Reconstructed spheres acquired along a trajectory are used as input for a robust dense spherical tracking algorithm which estimates the spheres' positions. Through the design of a new acquisition system it has been shown that it is possible to acquire these maps efficiently and a model has been provided for computing the augmented spherical representation. Furthermore, a calibration procedure has been developed that accounts for loop closure on the camera ring.

In perspective, since the method proposed in this paper deviates from classical 3D texture mapped models as well as classical panoramic spherical acquisition systems, many traditional tools are inadequate and need to be redesigned for the current system. Future effort will be aimed at improving divergent wide baseline matching (with large resolution differences between images) and taking into account illumination variation with differing aperture sizes between cameras around the camera ring.

VII. ACKNOWLEDGEMENTS

This work has been supported by ANR (French National Agency) CityVIP project under grant ANR-07_TSFA-013-01. The authors would like to thank Mathieu Seiler for helpful discussions and software development.

VIII. APPENDIX

A. Non-linear optimisation

The error functions for the calibration (7), the off-line graph learning (11) and the real-time tracking (13) are all minimized using a iteratively re-weighted least squared non-linear minimization:

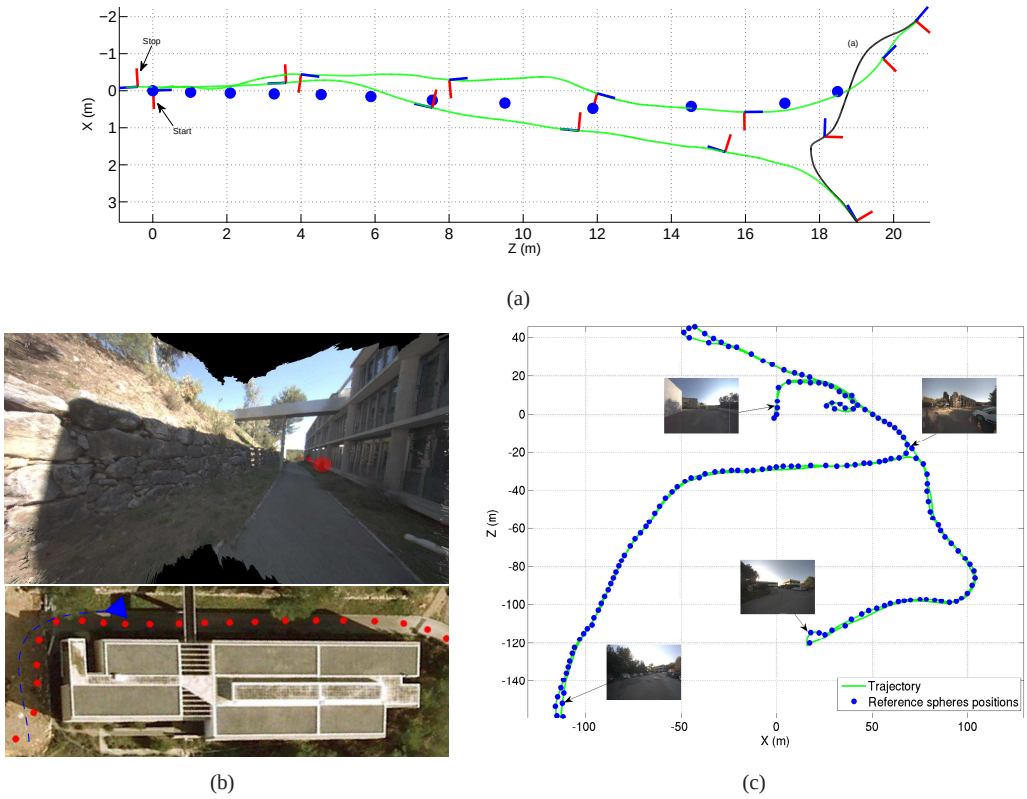
$$\mathcal{O}(\mathbf{x}) = \arg\min_x \sum_{i=1}^{6} \left(\mathbf{e}_i\right)^2, \tag{14}$$

by $\nabla\mathcal{O}(\mathbf{x})|_{\mathbf{x}=\tilde{\mathbf{x}}} = \mathbf{0}$, where ∇ is the gradient operator with respect to the unknown $\mathbf{x}$ defined in equation (2) assuming a global minimum is reached at $\mathbf{x} = \tilde{\mathbf{x}}$.

An inverse compositional algorithm is used [3], which allows to pre-compute most of the minimization parts directly on the reference image. In this case the unknown $\mathbf{x}$ is iteratively updated using a Levenberg-Marquart optimization procedure:

$$\mathbf{x} = -\lambda(\mathbf{Q} - \mu\mathrm{diag}(\mathbf{Q}))^{-1}\mathbf{J}^T\mathbf{D}\mathbf{e}, \tag{15}$$

where ^{T} is the transposition operator, $\mathbf{Q} = \mathbf{J}^T\mathbf{D}\mathbf{J}$ is the robust Gauss-Newton Hessian approximation, μ and λ are scalar gains to ensure a fast exponential error decrease. $\mathbf{J}$ is the warping Jacobian matrix of dimension $n \times 6$. $\mathbf{D}$ is a diagonal weighting matrix of dimension $n \times n$ obtained by M-estimation [13] which rejects outliers such as occlusions.



(a)

(b)

(c)

Fig. 6. (a) Real-time tracking of a monocular camera navigating within a portion of the graph containing 12 spheres (blue dots). The estimated trajectory is shown in green (the black part of the trajectory between two discontinuities is due to a forward-reverse movement of the mobile tracking system). Several camera poses are plotted (optical axis in blue) to show the orientation of the positioning system at various locations. (b) The top image shows a snapshot of our real-time interactive 3D OpenGL rendering platform which exploits the augmented spherical memory. It is possible to navigate freely in the virtual world with photo-realistic view synthesis. In both images, the red spheres indicate the 3D positions of the reference spheres in the world. The bottom image shows an aerial view of the mapped region and a real-time virtual camera trajectory is plotted in blue. (c) A 1.5 km reconstructed trajectory, after loop closures and graph optimization, with 310 reference spheres (one sphere out of two is plotted). Some key images are also displayed.

REFERENCES

[1] S. Avidan and A. Shashua. Novel view synthesis in tensor space. *In IEEE Conf. on Computer Vision and Pattern Recognition*, 0:1034, 1997.

[2] P. Baker, C. Fermuller, Y. Aloimonos, and R. Pless. A spherical eye from multiple cameras. *In IEEE Conf. on Computer Vision and Pattern Recognition*, 1:576, 2001.

[3] S. Baker and I. Matthews. Equivalence and efficiency of image alignment algorithms. In *IEEE Conf. on Computer Vision and Pattern Recognition*, volume 1, page 1090, 2001.

[4] P. J. Burt and E. H. Adelson. A multiresolution spline with application to image mosaics. *ACM Transactions on Graphics*, 2:217–236, 1983.

[5] G. Caron, E. Marchand, and E. Mouaddib. Tracking planes in omnidirectional stereovision. In *IEEE Int. Conf. on Robotics and Automation*, pages 6306–6311, 2011.

[6] D. Cobzas, H. Zhang, and M. Jagersand. Image-based localization with depth-enhanced image map. In *IEEE Conf. on Intelligent Robots and Systems*, pages 1570–1575, 2003.

[7] A.I. Comport, E. Malis, and P. Rives. Real-time quadrifocal visual odometry. *In The International Journal of Robotics Research*, 29(2-3):245–266, 2010.

[8] G. Gallegos, M. Meilland, P. Rives, and A.I. Comport. Appearance-based slam relying on a hybrid laser/omnidirectional sensor. In *IEEE Int. Conf. on Intelligent Robots and Systems*, pages 3005 –3010, 2010.

[9] C. Geyer and K. Daniilidis. Catadioptric projective geometry. *International Journal of Computer Vision*, 45:223–243, 2002.

[10] K. Hammoudi, F. Dornaika, B. Soheilian, and N. Paparoditis. Generating raw polygons of street facades from a 2d urban map and terrestrial laser range data. In *SSSI Australasian Remote Sensing and Photogrammetry Conf.*, 2010.

[11] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox. Rgb-d mapping: Using depth cameras for dense 3d modeling of indoor environments. In *Int. Symp. on Experimental Robotics*, 2010.

[12] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30:328–341, 2008.

[13] P.J. Huber. *Robust Statistics*. New york, Wiley, 1981.

[14] H.S. Kim and A. Hilton. Environment modelling using spherical stereo imaging. In *3DIM09*, pages 1534–1541, 2009.

[15] J. Kopf, B. Chen, R. Szeliski, and M Cohen. Street slide: Browsing street level imagery. *ACM Trans. on Graphics*, 29(4):96:1 – 96:8, 2010.

[16] G. Krishnan and S.K. Nayar. Towards A True Spherical Camera. In *SPIE Human Vision and Electronic Imaging*, 2009.

[17] S. Li. Full-view spherical image camera. *Int. Conf. on Pattern Recognition*, 4:386–390, 2006.

[18] M. Meilland, A.I. Comport, and P. Rives. A spherical robot-centered representation for urban navigation. In *IEEE Int. Conf. on Intelligent Robots and Systems*, pages 5196 –5201, 2010.

[19] M. Meilland, A.I. Comport, and P. Rives. Real-time direct model-based tracking under large lighting variations. In *British Machine Vision Conference*, 2011.

[20] S.K. Nayar. Catadioptric omnidirectional camera. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 482–, 1997.

[21] R. Szeliski. Image alignment and stitching: a tutorial. *Found. Trends. Comput. Graph. Vis.*, 2(1):1–104, 2006.

[22] A. Zaharescu, R. P. Horaud, R. Ronfard, and L. Lefort. Multiple camera calibration using robust perspective factorization. In *Int. Symp. on 3D Data Processing, Visualization and Transmission*, pages 504–511, 2006.